Computers and Mathematics with Applications 63 (2012) 1548–1556

Contents lists available at SciVerse ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

Numerical solution of unsteady Navier–Stokes equations on curvilinear meshes

Abdullah Shah [a,*], Li Yuan [b], Shamsul Islam [a]

[a] Department of Mathematics, COMSATS Institute of Information Technology, Park Road, Islamabad 44000, Pakistan

[b] LSEC and Institute of Computational Mathematics and Scientific/Engineering Computing, Academy of Mathematics & Systems Science, Chinese Academy of Sciences, Beijing 100190, PR China

ARTICLE INFO

Article history:
Received 26 September 2010
Received in revised form 2 December 2011
Accepted 15 March 2012

Keywords:
Incompressible Navier–Stokes equations
Artificial compressibility
Upwind compact scheme
Oscillating plate
Taylor decaying vortices
Flow over a cylinder

ABSTRACT

The objective of the present work is to extend our FDS-based third-order upwind compact schemes by Shah et al. (2009) [8] to numerical solutions of the unsteady incompressible Navier–Stokes equations in curvilinear coordinates, which will save much computing time and memory allocation by clustering grids in regions of high velocity gradients. The dual-time stepping approach is used for obtaining a divergence-free flow field at each physical time step. We have focused on addressing the crucial issue of implementing upwind compact schemes for the convective terms and a central compact scheme for the viscous terms on curvilinear structured grids. The method is evaluated in solving several two-dimensional unsteady benchmark flow problems.

1. Introduction

Mass conservation is the main issue in solving Incompressible Navier–Stokes equations (INSEs) numerically in the primitive variable form. Most numerical methods for INSEs require solving the pressure Poisson equation which is computationally expansive specially for three dimensional problems. In order to apply compressible flow solvers to incompressible problems, the mass conservation equation can be modified to couple it with the momentum equation. This goal can be achieved by adding pseudo-time derivative of pressure to the continuity equation [1] resulting in hyperbolic–parabolic type of system of equations. Thus, efficient algorithm like approximate-factorization scheme by Beam and Warming [2] originally developed for compressible flows, can be used to solve the incompressible flows.

The artificial compressibility (AC) method for unsteady flow computations has developed by many investigators including [3–7], who utilized without exception the dual-time stepping (DTS) technique [3]. In this technique, at each physical time step, pseudo-time marching (sub-iteration) is used to satisfy the mass conservation equation. Thus various discretization schemes and solution algorithms for compressible flow were used in conjunction with the AC method.

In the present paper, we extend our methodology [8] to the unsteady 2D INSEs by using DTS approach. The robustness of the method is apparent from the grid resolution by clustering grids in the region with high gradients. The validated cases are selected to demonstrate that the solution procedure is robust and accurate for complicated geometries with non-uniform grids.

The rest of the paper is organized as follows: In Section 2, formulations for the AC method in generalized curvilinear coordinates are briefly outlined. Section 3 presents the spatial discretization schemes. Section 4 describes the time-marching scheme for solving the discretized equations. The computed results are presented in Section 5. Finally, conclusions are drawn in Section 6.

* Corresponding author. Tel.: +92 321 5022236.
E-mail addresses: scholar.cm@gmail.com (A. Shah), lyuan@lsec.cc.ac.cn (L. Yuan).

0898-1221/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2012.03.047

2. Governing equations

We start from the artificial compressibility formulation of the unsteady 2D INSEs. In a general but stationary curvilinear coordinate (ξ, η), the transformed governing equations with the artificial compressibility term added to the continuity equation and the pseudo-time derivatives of the velocity added to the momentum equations for DTS are [5];

$$\frac{\partial \hat{\mathbf{Q}}}{\partial \tau} + \mathbf{I}_m \frac{\partial \hat{\mathbf{Q}}}{\partial t} + \frac{\partial(\hat{\mathbf{E}} - \hat{\mathbf{E}}_v)}{\partial \xi} + \frac{\partial(\hat{\mathbf{F}} - \hat{\mathbf{F}}_v)}{\partial \eta} = 0, \tag{2.1}$$

where

$$\hat{\mathbf{Q}} = \frac{\mathbf{Q}}{J} = \frac{1}{J}\begin{bmatrix} p \\ u \\ v \end{bmatrix}, \qquad \hat{\mathbf{E}} = \frac{1}{J}\begin{bmatrix} \beta U \\ uU + \xi_x p \\ vU + \xi_y p \end{bmatrix}, \qquad \hat{\mathbf{F}} = \frac{1}{J}\begin{bmatrix} \beta V \\ uV + \eta_x p \\ vV + \eta_y p \end{bmatrix},$$

$$\hat{\mathbf{E}}_v = \frac{1}{\mathrm{Re}J}\begin{bmatrix} 0 \\ (\xi_x^2 + \xi_y^2)u_\xi + (\xi_x\eta_x + \xi_y\eta_y)u_\eta \\ (\xi_x^2 + \xi_y^2)v_\xi + (\xi_x\eta_x + \xi_y\eta_y)v_\eta \end{bmatrix},$$

$$\hat{\mathbf{F}}_v = \frac{1}{\mathrm{Re}J}\begin{bmatrix} 0 \\ (\xi_x\eta_x + \xi_y\eta_y)u_\xi + (\eta_x^2 + \eta_y^2)u_\eta \\ (\xi_x\eta_x + \xi_y\eta_y)v_\xi + (\eta_x^2 + \eta_y^2)v_\eta \end{bmatrix},$$

$$U = \xi_x u + \xi_y v, \qquad V = \eta_x u + \eta_y v, \qquad \mathbf{I}_m = \mathrm{diag}(0, 1, 1),$$

τ is the pseudo-time, t is the physical time, p is the pressure, u and v are velocity components in Cartesian coordinates, β is the artificial compressibility factor, U and V are the contravariant velocity components in curvilinear coordinate directions, ξ and η, respectively, $\hat{\mathbf{E}}$ and $\hat{\mathbf{F}}$ are the inviscid flux vectors, $\hat{\mathbf{E}}_v$ and $\hat{\mathbf{F}}_v$ are the viscous flux vectors, Re is the Reynolds number, $\mathrm{Re} = \frac{U_\infty L}{\nu}$, based on a characteristic velocity U_∞ and a characteristic length L. Notice that in time-accurate numerical solutions to Eq. (2.1), the pseudo-time derivative term must be marched (iterated) to zero in order for the original INSEs to be satisfied, while the second term in Eq. (2.1) may be dropped for steady-state solutions. The Jacobian determinant of transformation J is defined as

$$J = \left|\frac{\partial(\xi, \eta)}{\partial(x, y)}\right| = \left|\frac{\partial(x, y)}{\partial(\xi, \eta)}\right|^{-1} = \begin{vmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{vmatrix}^{-1}.$$

Other metrics are

$$\frac{\xi_x}{J} = y_\eta, \qquad \frac{\xi_y}{J} = -x_\eta, \qquad \frac{\eta_x}{J} = -y_\xi, \qquad \frac{\eta_y}{J} = x_\xi. \tag{2.2}$$

The inviscid flux vectors for both ξ and η directions can be uniformly written as

$$\hat{\mathbf{E}}_k = \hat{k}_x\mathbf{E} + \hat{k}_y\mathbf{F} = \begin{bmatrix} \beta\Theta \\ u\Theta + \hat{k}_x p \\ v\Theta + \hat{k}_y p \end{bmatrix}. \tag{2.3}$$

where $\mathbf{E}$ and $\mathbf{F}$ are inviscid flux vectors in x and y directions, respectively, $\Theta = \hat{k}_x u + \hat{k}_y v$, $(\hat{k}_x, \hat{k}_y) = (k_x, k_y)/J$, $k = \xi, \eta$. The Jacobian matrix of this generalized flux vector is

$$\hat{\mathbf{A}}_k = \frac{\partial \hat{\mathbf{E}}_k}{\partial \mathbf{Q}} = \begin{bmatrix} 0 & \hat{k}_x\beta & \hat{k}_y\beta \\ \hat{k}_x & \hat{k}_x u + \Theta & \hat{k}_y u \\ \hat{k}_y & \hat{k}_x v & \hat{k}_y v + \Theta \end{bmatrix}. \tag{2.4}$$

A similarity transformation for the Jacobian matrix is

$$\hat{\mathbf{A}}_k = \mathbf{T}_k \boldsymbol{\Lambda}_k \mathbf{T}_k^{-1}, \tag{2.5}$$

with eigenvalue matrix $\boldsymbol{\Lambda}_k = \mathrm{diag}(\Theta, \Theta + c, \Theta - c)$, where c is the scaled artificial speed of sound given by

$$c = \sqrt{\Theta^2 + \beta(\hat{k}_x^2 + \hat{k}_y^2)}.$$

The right and left eigenvector matrices are $\mathbf{T}_k$ and $\mathbf{T}_k^{-1}$ respectively. The hyperbolic nature of the artificial compressibility formulation provides basis for the development of upwind schemes.

3. Spatial discretization

Owing to the hyperbolic nature of Eq. (2.1), the convective flux, and thus its derivative, can be split into two parts, e.g., in the ξ direction,

$$\hat{\mathbf{E}}_\xi = (\hat{\mathbf{E}}^+ + \hat{\mathbf{E}}^-)_\xi = \hat{\mathbf{E}}^+_\xi + \hat{\mathbf{E}}^-_\xi, \tag{3.1}$$

where subscript ξ denotes the partial derivative with respect to ξ, $\hat{\mathbf{E}}^+$ corresponds to the flux in the positive ξ direction with information being propagated by the positive eigenvalues, and $\hat{\mathbf{E}}^-$ corresponds to the flux in the negative ξ direction with information being propagated by the negative eigenvalues. To compute the split derivatives in Eq. (3.1), we use the third-order accurate upwind compact finite difference schemes developed by Fu et al. [9] and Fu and Ma [10], which are

$$\frac{2}{3}(\hat{\mathbf{E}}^+_\xi)_i + \frac{1}{3}(\hat{\mathbf{E}}^+_\xi)_{i-1} = \frac{5\delta^-\hat{\mathbf{E}}^+_i + \delta^-\hat{\mathbf{E}}^+_{i+1}}{6\Delta\xi}, \tag{3.2a}$$

$$\frac{2}{3}(\hat{\mathbf{E}}^-_\xi)_i + \frac{1}{3}(\hat{\mathbf{E}}^-_\xi)_{i+1} = \frac{5\delta^+\hat{\mathbf{E}}^-_i + \delta^+\hat{\mathbf{E}}^-_{i-1}}{6\Delta\xi}, \tag{3.2b}$$

where $\delta^+ f_i = f_{i+1} - f_i$, $\delta^- f_i = f_i - f_{i-1}$, and $\Delta\xi$ is the grid spacing in the computational space. Computation of the split derivatives in the left-hand side (LHS) is quite simple, as Eq. (3.2a) can be explicitly marched forward and Eq. (3.2b) backward to get all the derivatives as long as the right-hand side (RHS) and the boundary derivatives are given. The key is how to evaluate the RHS of Eq. (3.2). We note that each term in the RHS of Eq. (3.2) represents the difference of split fluxes between neighboring grid points. It can be shown from Eq. (2.3) that the difference of the original flux in ξ direction between neighboring grid points is

$$\begin{aligned}
\hat{\mathbf{E}}_{i+1} - \hat{\mathbf{E}}_i &= (\hat{\xi}_x\mathbf{E} + \hat{\xi}_y\mathbf{F})_{i+1} - (\hat{\xi}_x\mathbf{E} + \hat{\xi}_y\mathbf{F})_i \\
&= (\overline{\xi}_x \triangle \mathbf{E} + \overline{\xi}_y \triangle \mathbf{F}) + \overline{\mathbf{E}} \triangle \hat{\xi}_x + \overline{\mathbf{F}} \triangle \hat{\xi}_y \\
&= (\overline{\xi}_x\mathbf{A}(\overline{\mathbf{Q}}) \triangle \mathbf{Q} + \overline{\xi}_y\mathbf{B}(\overline{\mathbf{Q}}) \triangle \mathbf{Q}) + \overline{\mathbf{E}} \triangle \hat{\xi}_x + \overline{\mathbf{F}} \triangle \hat{\xi}_y \\
&= \hat{\mathbf{A}}_{i+\frac{1}{2}}(\overline{\mathbf{Q}}) \triangle \mathbf{Q} + \overline{\mathbf{E}} \triangle \hat{\xi}_x + \overline{\mathbf{F}} \triangle \hat{\xi}_y,
\end{aligned} \tag{3.3}$$

where $\triangle$ denotes the difference between points i and $i+1$, $\overline{\mathbf{Q}}$ is the arithmetic average of $\mathbf{Q}$ between points i and $i+1$, and we have utilized the exact Roe property in Cartesian coordinates [11], $\triangle\mathbf{E} = \mathbf{A}_{i+\frac{1}{2}}(\overline{\mathbf{Q}}) \triangle \mathbf{Q}$. The remaining two terms reflect the influence of varying grids (summing these terms in all directions leads to discretized geometric identities). We will drop the two terms as previous researchers did when implementing flux-difference splitting schemes on curvilinear grids [4]. Then the difference of the convective flux can be split based on the sign of the eigenvalues of the convective flux Jacobian matrix:

$$\hat{\mathbf{E}}^\pm_{i+1} - \hat{\mathbf{E}}^\pm_i \equiv \triangle\hat{\mathbf{E}}^\pm_{i+\frac{1}{2}} = \hat{\mathbf{A}}^\pm_{i+\frac{1}{2}}(\overline{\mathbf{Q}}) \triangle \mathbf{Q}. \tag{3.4}$$

The split Jacobian matrix is given by $\hat{\mathbf{A}}^\pm_{i+\frac{1}{2}}(\overline{\mathbf{Q}}) = \mathbf{T}_A\boldsymbol{\Lambda}^\pm_A\mathbf{T}_A^{-1}$, with $\boldsymbol{\Lambda}^\pm_A = \frac{1}{2}(\boldsymbol{\Lambda}_A \pm |\boldsymbol{\Lambda}_A|)$, which is evaluated using arithmetic averages of flow variables and metrics between i and $i+1$ points:

$$\overline{\mathbf{Q}} = \frac{1}{2}(\mathbf{Q}_i + \mathbf{Q}_{i+1}), \qquad (\overline{\xi}_x, \overline{\xi}_y) = \frac{1}{2}((\hat{\xi}_x, \hat{\xi}_y)_i + (\hat{\xi}_x, \hat{\xi}_y)_{i+1}). \tag{3.5}$$

To close the scheme, an explicit, dissipative, and third-order one-sided boundary scheme [12] is used at boundary grid points:

$$\begin{aligned}
\text{at } i = 1 : (\hat{\mathbf{E}}^+_\xi)_i &= \frac{-11\hat{\mathbf{E}}^+_i + 18\hat{\mathbf{E}}^+_{i+1} - 9\hat{\mathbf{E}}^+_{i+2} + 2\hat{\mathbf{E}}^+_{i+3}}{6\Delta\xi} \\
&= \frac{11 \triangle \hat{\mathbf{E}}^+_{i+\frac{1}{2}} - 7 \triangle \hat{\mathbf{E}}^+_{i+\frac{3}{2}} + 2 \triangle \hat{\mathbf{E}}^+_{i+\frac{5}{2}}}{6\Delta\xi},
\end{aligned} \tag{3.6a}$$

$$\begin{aligned}
\text{at } i = N : (\hat{\mathbf{E}}^-_\xi)_i &= \frac{11\hat{\mathbf{E}}^-_i - 18\hat{\mathbf{E}}^-_{i-1} + 9\hat{\mathbf{E}}^-_{i-2} - 2\hat{\mathbf{E}}^-_{i-3}}{6\Delta\xi} \\
&= \frac{11 \triangle \hat{\mathbf{E}}^-_{i-\frac{1}{2}} - 7 \triangle \hat{\mathbf{E}}^-_{i-\frac{3}{2}} + 2 \triangle \hat{\mathbf{E}}^-_{i-\frac{5}{2}}}{6\Delta\xi}.
\end{aligned} \tag{3.6b}$$

Evaluation of the viscous terms reduces to computing half-node second derivatives like $\partial_\xi(g\,\partial_\xi u)$, plus mixed derivative like $\partial_\xi(g\,\partial_\eta u)$ if non-orthogonal grids are used. It was recognized that discretization of second derivatives has higher

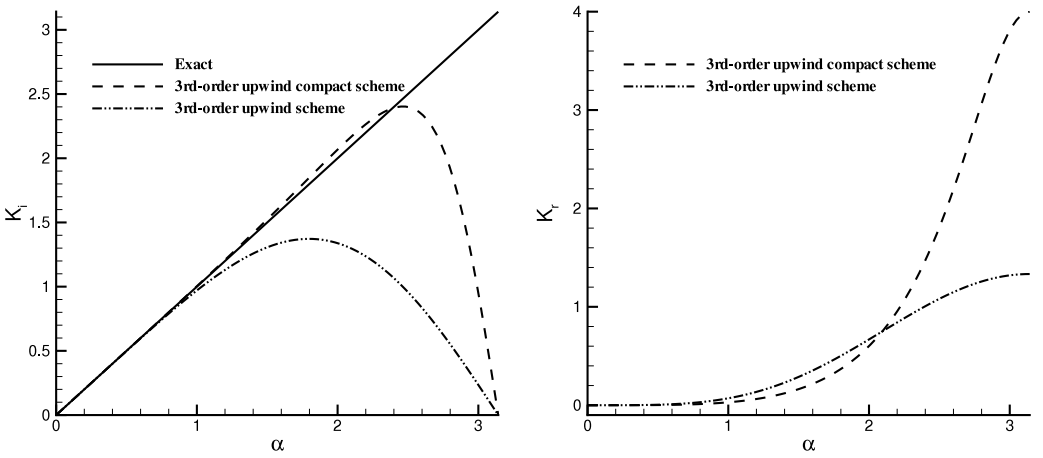

Fig. 1. Variations of k_r and k_i vs. α for the compact and noncompact schemes.

resolution than first derivatives, for this reason and to avoid large computational cost, explicit finite differences were often used [7] for the viscous terms.

To compare with Rogers and Kwak's upwind differencing scheme, we reformulate their third-order conservative scheme [5] as following nonconservative version:

$$(\hat{\mathbf{E}}_\xi)_i^{3\text{rd}} = \frac{1}{6\Delta\xi}\left[\left(2\Delta\hat{\mathbf{E}}^+_{i+\frac{1}{2}} + 5\Delta\hat{\mathbf{E}}^+_{i-\frac{1}{2}} - \Delta\hat{\mathbf{E}}^+_{i-\frac{3}{2}}\right) + \left(2\Delta\hat{\mathbf{E}}^-_{i-\frac{1}{2}} + 5\Delta\hat{\mathbf{E}}^-_{i+\frac{1}{2}} - \Delta\hat{\mathbf{E}}^-_{i+\frac{3}{2}}\right)\right]. \tag{3.7a}$$

In order to examine approximating behaviors of the two schemes, we use the following model equation and its semi-discrete approximation as follows:

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0, \quad c > 0, \tag{3.8a}$$

$$\frac{\partial u_j}{\partial t} + cF_j = 0, \tag{3.8b}$$

where F_j is finite difference approximation to $\partial u/\partial x$. With the initial condition $u(x, 0) = e^{ikx}$, the exact solution of Eq. (3.8a) is $u(x, t) = e^{ik(x-ct)}$, and the exact solution of difference equation. (3.8b) can be written as $u(x_j, t) = e^{-k_r \frac{ct}{\Delta x}} e^{ik(x_j - \frac{k_i}{k\Delta x}ct)}$, where the modified wave number $k_e = k_r + ik_i$ can be obtained from any given difference schemes [10]. k_i is related to the phase speed in the numerical solution, and k_r is related to the numerical damping of a difference scheme.

For the third-order upwind compact scheme Eq. (3.2a),

$$k_r = \frac{(1-\cos\alpha)^2}{5+4\cos\alpha}, \qquad k_i = \frac{\sin\alpha(8+\cos\alpha)}{5+4\cos\alpha}, \tag{3.9}$$

where the reduced wave number $\alpha = k\Delta x$. For the explicit third-order upwind scheme Eq. (3.7a),

$$k_r = \frac{1}{6}(3 - 4\cos\alpha + \cos 2\alpha), \qquad k_i = \frac{1}{6}(8\sin\alpha - \sin 2\alpha). \tag{3.10}$$

Fig. 1 shows variations of k_r and k_i with the reduced wave number α for the above two schemes. We can see the third-order upwind compact scheme can approximate the exact damping ($k_r^E = 0$) to higher waver number, and approximate the exact dispersion relation ($k_i^E = \alpha$) better than the noncompact scheme.

Table 1 gives the upper limit of the reduced wave number, which corresponds to a point in Fig. 1 where k_r or k_i begins to reach 2% errors relative to their exact solutions respectively. To approximate the exact wave speed within 2% error, the ratio of grid points needed by the third-order upwind compact scheme to those needed by the third-order upwind scheme is $0.902/1.61 = 0.56$ in one dimensional and $(0.902/1.61)^3 = 0.18$ in three-dimensional case, resulting in significant saving in computer resources.

4. Implicit approximate factorization

Applying backward Euler's scheme to the pseudo-time, and second-order backward difference scheme to the physical-time derivative in Eq. (2.1), we have

$$\frac{\Delta\mathbf{Q}^{n+1,m}}{J\Delta\tau} + \mathbf{I}_m\frac{1.5\mathbf{Q}^{n+1,m+1} - 2\mathbf{Q}^n + 0.5\mathbf{Q}^{n-1}}{J\Delta t} + \left[\frac{\partial(\hat{\mathbf{E}} - \hat{\mathbf{E}}_v)}{\partial\xi} + \frac{\partial(\hat{\mathbf{F}} - \hat{\mathbf{F}}_v)}{\partial\eta}\right]^{n+1,m+1} = 0, \tag{4.1}$$

Table 1
Upper limits of the reduced wave number when k_r and k_i of the difference schemes first exceed 2% errors relative to their exact solutions.

Scheme	Upper limits of wave number	
	$k_r < 2\%$	$\lvert 1 - k_i/\alpha\rvert < 2\%$
Third-order upwind compact	0.91	1.61
Third-order upwind	0.72	0.902

where $\Delta\mathbf{Q}^{n+1,m} = \mathbf{Q}^{n+1,m+1} - \mathbf{Q}^{n+1,m}$ is the increment or correction of conservative variables, superscript n denotes the physical time level, and m denotes the pseudo-time level (number of sub-iterations). $\Delta\tau$ is the pseudo-time step which is determined based on CFL number and Δt is the physical time step. For time-accurate solutions, Eq. (4.1) is iterated in pseudo-time so that the transient $\mathbf{Q}^{n+1,m+1}$ approaches the physical $\mathbf{Q}^{n+1}$ as the first term approaches zero. To derive an implicit method, all terms at $m+1$ pseudo-time level are linearized with respect to the previous level m by using Taylor expansion, e.g.

$$\hat{\mathbf{E}}^{m+1} \approx \hat{\mathbf{E}}^m + \left(\frac{\partial\hat{\mathbf{E}}}{\partial\mathbf{Q}}\right)^m \Delta\mathbf{Q}^m = \hat{\mathbf{E}}^m + \hat{\mathbf{A}}^m\Delta\mathbf{Q}^m + O(\|\Delta\mathbf{Q}^m\|^2) \tag{4.2a}$$

$$\hat{\mathbf{E}}_v^{m+1} \approx \hat{\mathbf{E}}_v^m + \left(\frac{\partial\hat{\mathbf{E}}_v}{\partial\mathbf{Q}}\right)^m \Delta\mathbf{Q}^m = \hat{\mathbf{E}}_v^m + \hat{\mathbf{A}}_v^m\Delta\mathbf{Q}^m + O(\|\Delta\mathbf{Q}^m\|^2) \tag{4.2b}$$

where superscript $n+1$ is omitted for brevity. Then we obtain the linearized, unfactored implicit delta form as

$$\begin{aligned}\left[\mathbf{I} + 1.5\frac{\Delta\tau}{\Delta t}\mathbf{I}_m + \Delta\tau J\left(\frac{\partial(\hat{\mathbf{A}} - \hat{\mathbf{A}}_v)}{\partial\xi} + \frac{\partial(\hat{\mathbf{B}} - \hat{\mathbf{B}}_v)}{\partial\eta}\right)\right]^m \Delta\mathbf{Q}^m &= -\Delta\tau J\left(\frac{\partial(\hat{\mathbf{E}} - \hat{\mathbf{E}}_v)}{\partial\xi} + \frac{\partial(\hat{\mathbf{F}} - \hat{\mathbf{F}}_v)}{\partial\eta}\right)^m \\ &\quad - \frac{\Delta\tau}{\Delta t}\mathbf{I}_m(1.5\mathbf{Q}^m - 2\mathbf{Q}^n + 0.5\mathbf{Q}^{n-1}) \\ &= -\hat{\mathbf{R}}^m.\end{aligned} \tag{4.3}$$

For factorization purpose, we will only retain the orthogonal parts in the viscous Jacobian matrices $\mathbf{A}_v$ and $\mathbf{B}_v$ in Eq. (4.3). The orthogonal parts are

$$\mathbf{A}'_v = \left(\frac{1}{\mathrm{Re}J}\,\nabla\xi\cdot\nabla\xi\right)\mathbf{I}_m\frac{\partial}{\partial\xi} = \gamma_1\mathbf{I}_m\frac{\partial}{\partial\xi}, \tag{4.4a}$$

$$\mathbf{B}'_v = \left(\frac{1}{\mathrm{Re}J}\,\nabla\eta\cdot\nabla\eta\right)\mathbf{I}_m\frac{\partial}{\partial\eta} = \gamma_2\mathbf{I}_m\frac{\partial}{\partial\eta}. \tag{4.4b}$$

For a steady-state computation only, terms containing Δt are dropped in Eq. (4.3). In general cases, denote $\mathbf{D} = \mathbf{I} + 1.5\frac{\Delta\tau}{\Delta t}\mathbf{I}_m$. Then the left-hand side operator of Eq. (4.3) can be approximately factorized to obtain the Beam–Warming scheme [2] as

$$\left[\mathbf{D} + \Delta\tau J\frac{\partial(\hat{\mathbf{A}} - \mathbf{A}'_v)}{\partial\xi}\right]^m \mathbf{D}^{-1}\left[\mathbf{D} + \Delta\tau J\frac{\partial(\hat{\mathbf{B}} - \mathbf{B}'_v)}{\partial\eta}\right]^m \Delta\mathbf{Q}^m = -\hat{\mathbf{R}}^m. \tag{4.5}$$

To obtain block tridiagonal equations, the convective terms in the left-hand side of Eq. (4.5) are discretized by the first-order upwind difference and viscous terms by the second-order central difference, e.g.,

$$\partial_\xi\hat{\mathbf{A}} = \delta_\xi^+\hat{\mathbf{A}}^- + \delta_\xi^-\hat{\mathbf{A}}^+, \qquad \partial_\xi\mathbf{A}'_v = \partial_\xi\gamma_1\mathbf{I}_m\partial_\xi$$

where $\delta_\xi^\pm f = \frac{\pm(f_{i\pm1} - f_i)}{\Delta\xi}$, $\delta_\xi\gamma_1\delta_\xi f = \frac{\gamma_{1_{i+1/2}}(f_{i+1} - f_i) - \gamma_{1_{i-1/2}}(f_i - f_{i-1})}{\Delta\xi^2}$. Notice that high-order compact schemes are still used for the residual, $\hat{\mathbf{R}}^m$. Thus we obtain the Approximate Factorization scheme in the form

$$[\mathbf{D} + \Delta\tau J(\delta_\xi^-\hat{\mathbf{A}}^+ + \delta_\xi^+\hat{\mathbf{A}}^- - \delta_\xi\gamma_1\mathbf{I}_m\delta_\xi)]^m\mathbf{D}^{-1}[\mathbf{D} + \Delta\tau J(\delta_\eta^-\hat{\mathbf{B}}^+ + \delta_\eta^+\hat{\mathbf{B}}^- - \delta_\eta\gamma_2\mathbf{I}_m\delta_\eta)]^m\Delta\mathbf{Q}^m = -\hat{\mathbf{R}}^m, \tag{4.6}$$

which requires solving block-tridiagonal equations in ξ and η directions alternatively. Dirichlet boundary conditions, $\Delta\mathbf{Q}_1 = \Delta\mathbf{Q}_{\mathrm{imax}} = 0$, are always used except at periodic boundaries where a periodic tridiagonal solver is used.

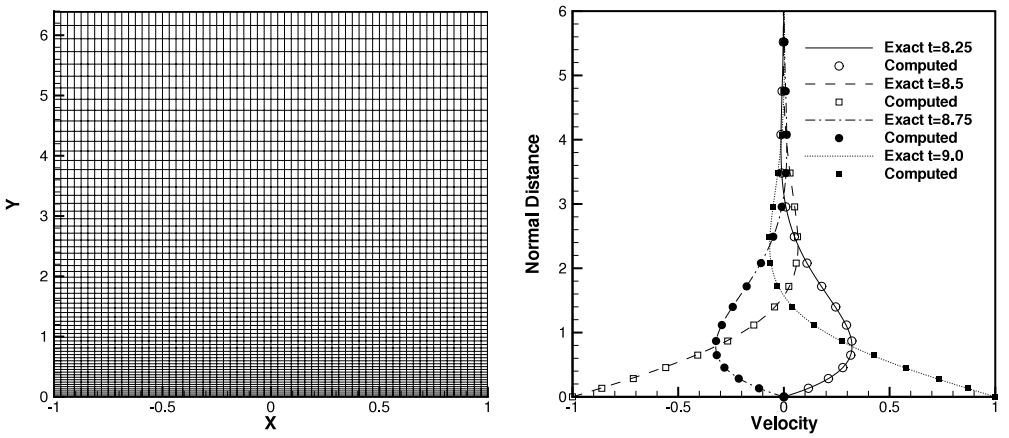

Fig. 2. a. Stretching grid b. Comparison of the exact (lines) with the computed results (symbols).

5. Numerical results

In this section, the numerical method developed in previous sections is tested for solving unsteady flow problems.

5.1. Oscillating Plate

The first test problem is the flow over an infinite oscillating plate[4]. The geometry of the problem is such that the x-axis is set along the plate and y-axis normal to it. The velocity of the plate is given by;

$$u_{plate} = u_0 \cos \omega t.$$

The exact solution for this problem is

$$u(y, t) = u_0 \exp(-ky) \cos(\omega t - ky),$$

with $k = \sqrt{\frac{\omega}{2\upsilon}}$. For the present computation, the initial velocity u_0 and the constant k was set to unity. Setting the frequency to 2π so that $\upsilon = \pi$. The computational grid used here is 65 points distributed with a simple exponential stretching $y = e^{\eta} - 1$ normal to the wall direction. The physical time step Δt was set to 0.01 and the was run over nine flow cycles after which the transient solution had died and time-periodic response was reached. In Fig. 2, comparison of the computed velocity distribution with the exact solution at different time during the cycle are plotted. The agreement of computed solution with the exact solution is very good.

5.2. The Taylor decaying vortex problem

This is a benchmark problem for validating unsteady flow solver [13,14] for the INSEs. The initial condition to the problem in square domain $0 \leq x, y \leq 2\pi$ is

$$u(x, y, 0) = -\cos(x) \sin(y),$$
$$u(x, y, 0) = \sin(x) \cos(y),$$
$$p(x, y, 0) = -0.25(\cos 2x + \cos 2y).$$

The initially periodical vortex structure convected by the flow-field and exponentially decaying due to the viscous decaying. The exact solution of the problem is

$$u(x, y, t) = -\cos(x) \sin(y) \exp(-2t/Re)$$
$$v(x, y, t) = \cos(y) \sin(x) \exp(-2t/Re)$$
$$p(x, y, t) = -0.25(\cos 2x + \cos 2y) \exp(-4t/Re).$$

In order to resolve the flow accurately, we have generated the grid in such a way (see Fig. 3(a)) that maximum number of points get allocated to those regions with large velocity gradient. The following stretching functions was used to generate the grid [15]:

$$x_i = \frac{\pi}{2} \left\{ \frac{i}{i_{\max}} + \frac{\lambda}{\pi} \sin\left(\frac{\pi i}{i_{\max}} \right) \right\} \quad \text{in } 0 \leq x, y \leq \frac{\pi}{2},$$

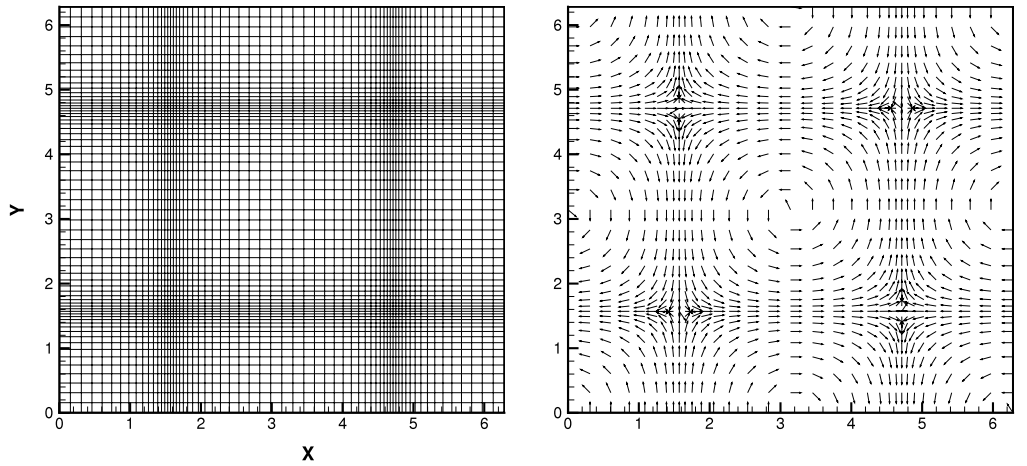

Fig. 3. Taylor's decaying vortex problem. (a) Computational grid (b) computed vector-field at $t = 2.0$.

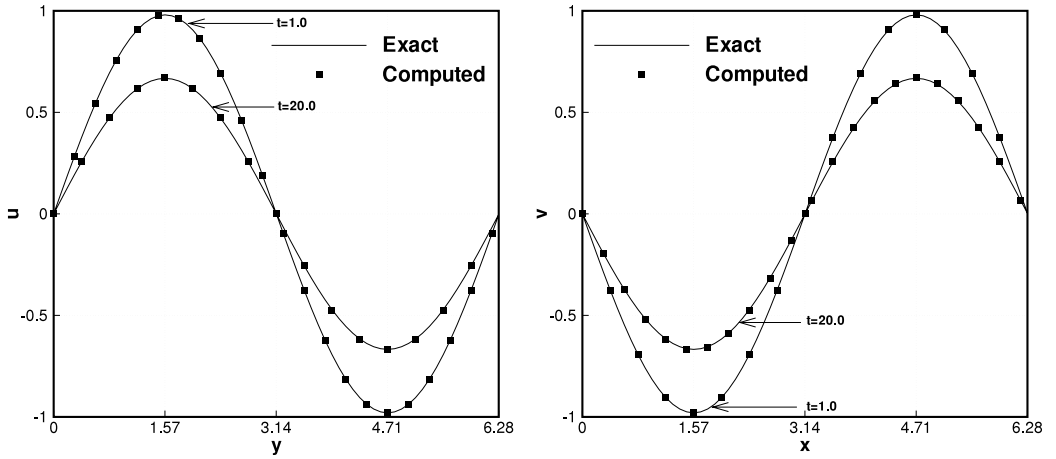

Fig. 4. The u velocity component along vertical centerline and the v velocity component along horizontal centerline for Taylor's decaying problem at 65×65 grid points.

$$x_i = \pi \left\{ \frac{i}{i_{\max}} - \frac{\lambda}{2\pi} \sin\left(\frac{2\pi i}{i_{\max}}\right) \right\} \quad \text{in } \frac{\pi}{2} \le x, y \le \frac{3\pi}{2},$$

$$x_i = \frac{\pi}{2} \left\{ \frac{i}{i_{\max}} - \frac{\lambda}{\pi} \sin\left(\frac{\pi i}{i_{\max}}\right) \right\} \quad \text{in } \frac{3\pi}{2} \le x, y \le 2\pi\,.$$

The parameter λ determine the degree of clustering with greater value of lambda giving more number of points at specified locations. We present our results computed on a 65×65 grid with $\lambda = 0.6$, $\Delta t = 0.05$, $\beta = 100$ and for $Re = 100$. Fig. 3(b) shows the computed vector field of the problem at $t = 2.0$ which is qualitatively comparable with the exact one.

The u velocity along vertical lines and the v velocity along horizontal lines through the geometric center are symmetric and have the same maximum amplitude as shown in Fig. 4. The computed results are in very good agreement with exact solutions.

5.3. Unsteady flow over a circular cylinder

The accuracy of the methods was investigated for an unsteady external flow over a two-dimensional circular cylinder at two different Reynolds numbers of 40 and 200 based on the diameter of the cylinder. Boundary conditions are no-slip on the cylinder and characteristic boundary conditions to treat the far-field. The computational grid used is 101×120 O-type with a radial cutting line in the upstream zone and with points clustered near the cylinder. Time step Δt was set to 0.01 in order to have converged solution, and pseudo-time step $\Delta\tau$ is determined based on a CFL number of 5. The maximum number of sub-iteration is set to 100. The artificial compressibility factor is set to 200. For $6 < Re < 40$, a steady state solution exists with a pair of symmetric separation bubbles on the leeward side while at higher Reynolds numbers the flow field become unsteady [16]. The time evolution of the separation bubble length for $Re = 40$ is compared with experimental values of

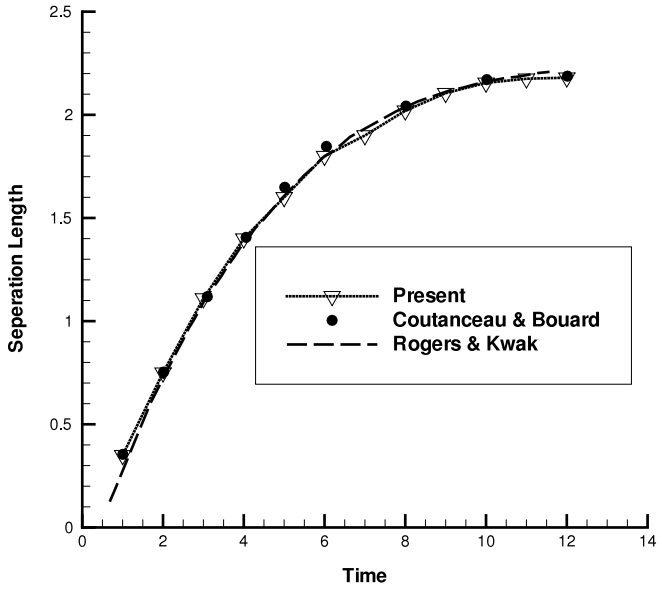

Fig. 5. Growth of separation length versus time behind a circular cylinder at $Re = 40$.

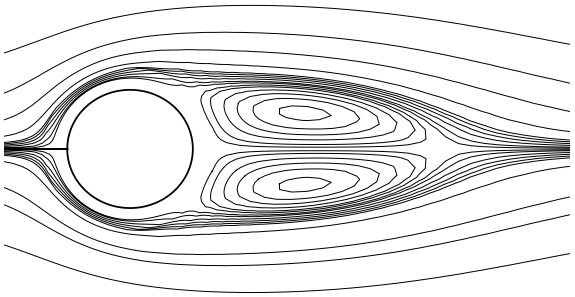

Fig. 6. The stream lines for flow over a circular cylinder at $Re = 40$ at $t = 12$.

Table 2
Lift and drag coefficients for flow past circular cylinder at $Re = 200$.

	C_l	C_d
Present	± 0.65	1.33 ± 0.05
Roger and Kwak [4]	± 0.75	1.29 ± 0.05
Y-N Chen et al. [17]	± 0.72	1.33 ± 0.04

Coutanceau and Bouard [16] and simulation results of Roger and Kwak [5] as shown in Fig. 5. The present results show good agreement. The corresponding stream lines for $t = 12$ is shown in Fig. 6.

The second problem for the same geometry is the case of Reynolds number of 200 based on diameter. The lift and drag coefficient are plotted versus time in Fig. 7. In Table 2, the lift and drag coefficient of the periodic state are listed which appear to be quite consistent with the existing results. It is clear that the periodicity of flow-field has been successfully revealed in the present calculations.

6. Conclusions

Based on artificial compressibility and dual-time stepping approach, an implicit third-order upwind compact scheme is developed for unsteady incompressible Navier–Stokes equations in curvilinear coordinates. Formulas for the compact scheme, their boundary scheme and their combination with the Beam–Warming approximate factorization are presented in detail. The resulting algorithm is tested on 2D benchmark flow problems with low to moderate Reynolds numbers. The numerical scheme is simple and capable to handle problems of complicated geometries using non-uniform grids.

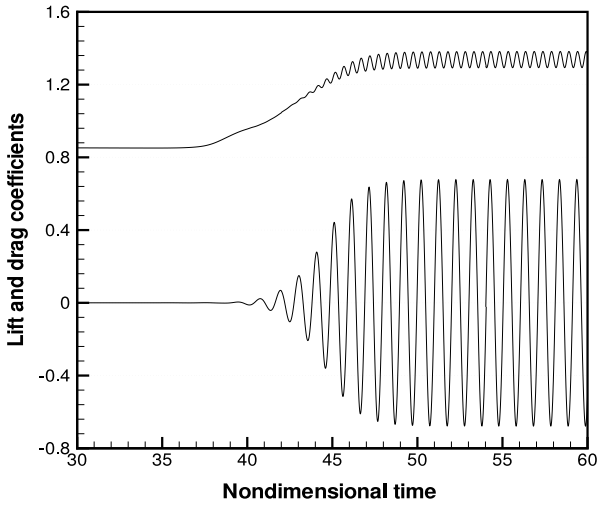

Fig. 7. Lift and drag coefficients vs. time for flow over a circular cylinder at $Re = 200$.

References

[1] A.J. Chorin, A numerical method for solving incompressible viscous flow problems, J. Comput. Phys. 2 (1967) 12–26.
[2] R. Beam, R.F. Warming, An implicit scheme for the compressible Navier–Stokes equations, AIAA J 16 (1978) 393–402.
[3] C.L. Mercle, M. Athavale, Time-accurate unsteady incompressible flow algorithm based on artificial compressibility method, AIAA Paper 87, 1137, 1987.
[4] S.E. Rogers, D. Kwak, An upwind differencing scheme for the time-accurate incompressible Navier–Stokes equations, AIAA J. 28 (2) (1990) 253–262.
[5] S.E. Rogers, D. Kwak, An upwind differencing scheme for the incompressible Navier–Stokes equations, Appl. Numer. Math. 8 (1991) 43–64.
[6] Li. Yuan, Comparison of implicit multigrid schemes for three dimensional incompressible flows, J. Comput. Phys. 177 (2002) 134–155.
[7] J. Ekarterinaris, High-order accurate numerical solutions of incompressible flows with the artifical compressibility method, Int. J. Numr. Meth. Fluids 45 (2004) 207–1187.
[8] Abdullah Shah, Hong Guo, Li Yuan, A third-order upwind compact scheme on curvilinear meshes for the incompressible Navier–Stokes equations, Commun. Comput. Phys. 5 (2–4) (2009) 712–729.
[9] D.X. Fu, Y.W. Ma, T. Kobayashi, Nonphysical oscillations in numerical solutions: reason and improvement, CFD J. 4 (4) (1996) 427–450.
[10] D.X. Fu, Y.W. Ma, A high order accurate finite difference scheme for complex flow fields, J. Comput. Phys. 134 (1997) 1–15.
[11] P.L. Roe, Approximate Riemann solvers, parameter vectors, and difference scheme, J. Comput. Phys. 43 (1981) 357–372.
[12] Y.Q. Shen, G.W. Yang, Z. Gao, High-resolution finite compact differences for hyperbolic conservation laws, J. Comput. Phys. 216 (2006) 114–137.
[13] J. Kim, P. Moin, Application of a fractional step methods to incompressible Navier Stokes equations, J. Comput. Phys. 59 (1985) 308–323.
[14] A. Quarteroni, F. Saleri, A. Veneziani, Factorization methods for the numerical approximation of Navier–Stokes equations, Comput. Methods Appl. Mech. Engrg. 188 (2000) 505–526.
[15] J.C. Kalita, A.K. Dass, Nimisha Nidhi, An efficient transient Navier–Stokes solver on compact nonuniform space grids, J. Comput. Appl. Math. 214 (2008) 148–162.
[16] M. Coutanceau, R. Bouard, Experimental determination of the main features of the viscous flow in the wake of a circular cylinder in uniform translation. Part 2. Unsteady flow, J. Fluid Mech. 79 (1977) 257–272.
[17] Y.N. Chen, S.C. Yang, J.Y. Yang, Implicit weighted essentially non-oscillatory schemes for the incompressible Navier–Stokes equations, Int. J. Numer. Meth. Fluids 31 (1999) 747–765.